Tutorial

Quality and Reliability Engineering International

(wileyonlinelibrary.com) DOI: 10.1002/qre.1924

Published online in Wiley Online Library

A Tutorial on Latin Hypercube Design of Experiments

Felipe A. C. Viana*†

The growing power of computers enabled techniques created for design and analysis of simulations to be applied to a large spectrum of problems and to reach high level of acceptance among practitioners. Generally, when simulations are time consuming, a surrogate model replaces the computer code in further studies (e.g., optimization, sensitivity analysis, etc.). The first step for a successful surrogate modeling and statistical analysis is the planning of the input configuration that is used to exercise the simulation code. Among the strategies devised for computer experiments, Latin hypercube designs have become particularly popular. This paper provides a tutorial on Latin hypercube design of experiments, highlighting potential reasons of its widespread use. The discussion starts with the early developments in optimization of the point selection and goes all the way to the pitfalls of the indiscriminate use of Latin hypercube designs. Final thoughts are given on opportunities for future research. Copyright © 2015 John Wiley & Sons, Ltd.

Keywords: design and analysis of computer experiments; Latin hypercube sampling; space-filling designs; sequential sampling

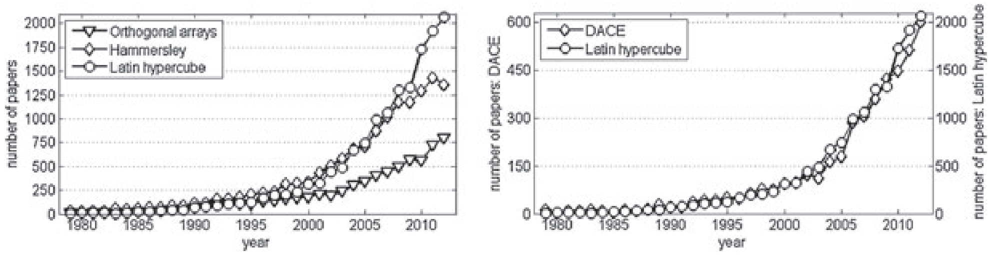
1. Introduction

Computer models usually replace physical experiments in studies like sensitivity analysis, design optimization, and reliability assessment. Frequently, there is very limited previous knowledge about the system (particularly in situations like engineering conceptual design), and analysts, scientists, and engineers tend to explore large input spaces. To make matters worse, in many cases, high-fidelity models are computationally expensive. Fortunately, years of research in mathematical formulation leveraged by the growth of computer power enabled techniques devised for design and analysis of simulations[1–5] to be successfully applied to a variety of problems (e.g., design of energy and aerospace[6–8] systems, manufacturing,[9] bioengineering,[10,11] and decision under uncertainty).[12] Such techniques embrace the set of methodologies for generating a surrogate model (also known as metamodel or response surface approximation), which is used to replace the expensive simulation code. The goal is constructing an approximation of the response that is as accurate as possible under limited number of expensive simulations.

With that said, careful planning of the inputs for the computer codes is first and one of the most crucial steps for a successful statistical modeling of the simulations. Often, few statistical assumptions are made about the input/output relationship of computer models. That might be one of the reasons why the initial sample is planned to cover most of the considered domain (leading to space-filling experimental designs). This is clearly elucidated in the vast literature about experimental designs for computer experiments.[13–17] Latin hypercube designs[18,19] have become particularly popular among strategies for computer experiments. Other strategies include orthogonal arrays[20] and Hammersley designs.[21,22] To illustrate their popularity, Figure 1(a) shows an approximate number of publications that referred to at least one of these three techniques. The data was obtained using the Google Scholar (http://scholar.google.com) database. While the specific numbers may vary, as the Google Scholar database is updated, there is stronger growth in the number of papers using either Hammersley or Latin hypercube designs when compared with orthogonal arrays. Figure 1(b) illustrates the close relationship between the growth in publications related to the design of computer experiments and Latin hypercube design. Another evidence of popularity is the number and diversity of the reported applications in which the Latin hypercube design is used. For example, the book edited by Dr. Koziel and Dr. Leifsson[23] is dedicated to applications of surrogate modeling, and the Latin hypercube design appears in 8 out of the 16 chapters. The applications range from circuit design to microwave structures to aerodynamic shape optimization.

This paper aims at providing a short overview of the research in Latin hypercube design of experiments with some hypotheses to explain its extensive use. Given that Latin hypercube designs can create samples that poorly cover the input domain, optimization of the Latin hypercube is first discussed. New practitioners would find here some explanations for why peers recommend them to use Latin hypercube designs. Then, the pitfalls of always using Latin hypercube designs for selecting experimental designs are

Probabilistics Laboratory, GE Global Research, Niskayuna, NY, USA
**Correspondence to: Felipe A. C. Viana, Probabilistics Laboratory, GE Global Research, Niskayuna, NY, USA.*
†E-mail: viana@ge.com



(a) Orthogonal arrays, Hammersley series, and Latin hypercube design.

(b) DACE versus Latin hypercube related.

Figure 1. Number of papers published per year. Data obtained from the Google Scholar (http://scholar.google.com) database in the week of March 4, 2013. For the sampling schemes, the search was set with 'any of these words': 'design of experiments' or 'experimental design,' or 'sampling' plus 'orthogonal arrays', 'Hammersley', or 'Latin hypercube'. For design and analysis of computer experiments (DACE), the search was set 'any of these words': 'design of computer experiments' or 'design of simulation experiments' or 'design and analysis of computer experiments'

highlighted. Finally, the research in Latin hypercube designs is situated in the current state of the art, and opportunities for future work are presented.

The remaining of the paper is organized as follows. Section 2 presents some general discussion about the Latin hypercube experimental design. Section 3 discusses five points of interest when researchers use such designs. Section 4 closes the paper recapitulating the important points and conclusions.

2. Latin hypercube and other experimental designs

Here, an experimental design with p points in d dimensions is written as a $p \times d$ matrix $\boldsymbol{X}=[\boldsymbol{x}_1\ \boldsymbol{x}_2 \ldots \boldsymbol{x}_p]^T$, where each column represents a variable, and each row $\boldsymbol{x}_i = \left[x_i^{(1)}\ x_i^{(2)}\ \ldots\ x_i^{(d)}\right]$ represents a sample. A Latin hypercube design is constructed in such a way that each of the d dimensions is divided into p equal levels (sometimes called bins) and that there is only one point (or sample) at each level. As originally proposed, a random procedure is used to determine the point locations. Figure 2 shows three examples of Latin hypercube designs with $d=2$ and $p=20$. The extreme case illustrated in Figure 2(a) is a Latin hypercube with very poor space filling qualities. Randomization alone could improve the experimental design to the point exemplified by Figure 2(b). On the other hand, optimization of point placement (as discussed in Section 3.2) would lead to the better choice shown in Figure 2(c), where points are more uniformly distributed over the domain.

There are other sampling techniques suitable for computer experiments.[‡] For example, orthogonal arrays organize the design matrix $\boldsymbol{X}$ in $p \times d$ matrix of integers $0 \leq x_{ij} \leq b-1$. The array is said to have strength $t \leq d$ if in every p by t submatrix of $\boldsymbol{X}$, all of the b^t possible rows appear the same number of λ times ($p=\lambda b^t$). Latin hypercube sampling corresponds to strength $t=1$, with $\lambda=1$. Hammersley designs are based on Hammersley sequences. Much like Fibonacci series, the Hammersley sequences are built using operations on integer numbers. For further reading on these three sampling schemes, please refer to.[21–24]

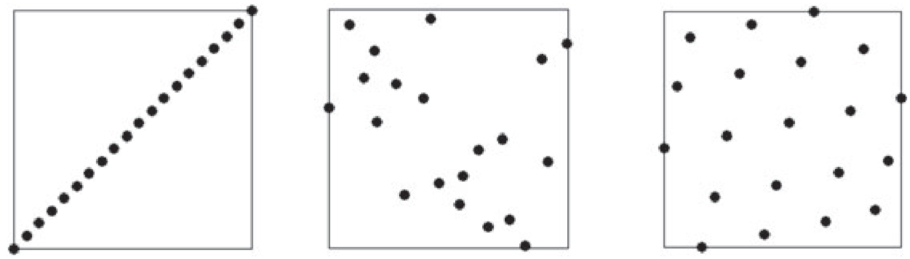
3. Five questions that make you think

This section presents an attempt to answer five intriguing questions about the Latin hypercube designs. By no means, the answers should be seen as definitive. Instead, they represent a partial (and why not say, personal) observation based on recent literature. The objective is trying to understand why Latin hypercube sampling is so popular, how much progress research has made, what the limitations are, what the alternatives are, and what remains to be performed.

3.1. Why do people like the Latin hypercube design so much?

In the early days of design and analysis of computer experiments, the practitioners were using both sampling and modeling techniques developed for physical experiments (see the appendix for brief discussion on physical and computer experiments). The maturity of design and analysis of computer experiments as a discipline opened up the following question: can sampling and modeling strategies be designed for computer experiments? Ideally, both the experimental design and the modeling strategies

[‡]Classical experimental designs, such as central composite designs or D-optimal designs, have been used for simulations (obviously concepts such as randomization, blocking, and replication might not make sense. For deterministic simulations, for example, replication is never applied). In fact, in the early days of computer experiments, practitioners relied on traditional response surface theory (for sampling and modeling) when dealing with simulations. Nevertheless, this practice has proven to be sub-efficient and, these days, less and less frequent.



(a) Design with very poor space filling properties.

(b) Randomized design.

(c) Design with good space filling properties.

Figure 2. Examples of Latin hypercube designs with $d = 2$ dimensions and $p = 20$ points

(not the topic here) would be developed to address the particularities of computer experiments.§ Classical experimental designs meant to deal with the non-deterministic and relatively low dimensional nature of physical experiments.¶ In such cases, concepts like replication and randomization (discussed in the appendix) make sense, and model assumptions guide the point placement. For computer experiments, however, an attractive sampling technique would have to be flexible enough to (i) provide data for modeling techniques based on very different statistical assumptions and (ii) be capable of covering small to large design spaces (no constraints in terms of data density and location). When Mckay *et al.* [18] and Iman and Conover [19] proposed the Latin hypercube design, they might not have foreseen the fact that the Latin hypercube design offers both, which would contribute to the popularity of the strategy. Although there might be other reasons for their popularity, the fact is that Latin hypercube designs can be optimized without having to look at the statistical assumptions of the model and have as many points as one can afford. From a practical standpoint, this makes the strategy very attractive.

Curiously, the Latin hypercube popularity might also have something to do with the historically difficult problem of optimizing designs for sophisticated modeling approaches such as the Gaussian process. The seminal paper by Sacks *et al.* [1] has a section devoted to the designs for computer experiments. After introducing the Gaussian process as modeling approach, the authors discussed criteria for a fixed number of runs and specified covariance structure, such as integrated mean squared error, maximum mean square error, and entropy. They acknowledge the difficulties in solving the resulting optimization problems. Much of that discussion holds true even today.

Another good reason for the Latin hypercube popularity is flexibility. For example, if few dimensions have to be dropped out, the resulting design is still a Latin hypercube design (maybe sub-optimal, but a Latin hypercube nevertheless). That happens because Latin hypercube samples are non-collapsing. Figure 3(a) illustrates the case of a Latin hypercube design with $d = 3$ dimensions and $p = 15$ points. Any of the two-dimensional projections is still a Latin hypercube design with the same $p = 15$ points (although, for this particular case, the $x_1 - x_2$ projection is the best in terms of space filling). Thus, if one cannot afford another set of data properly designed for the smaller domain, the existing data can be reused without reduction in number of sampled points. This is not the case when central composite or factorial designs are used. In such cases, once some of the dimensions are eliminated, points collapse one into another (reducing the sample size). Figure 3(b) shows the case of a circumscribed central composite design with $d = 3$ dimensions (resulting in the same $p = 15$ points). While the two-dimensional projections are still central composite designs, points collapse into one another, and the effective design size reduces to $p = 9$ points.

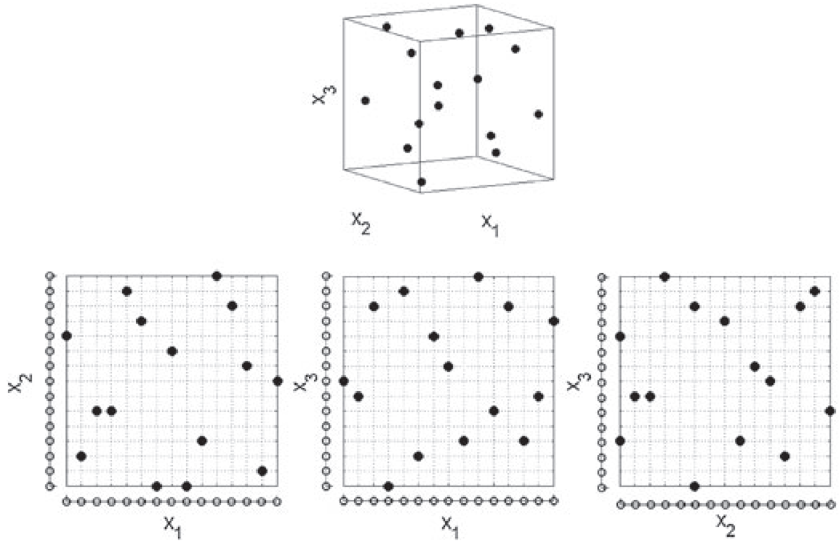
Unfortunately, the same thought process would not be true for the case of adding input variables to the problem. That is, adding one or more dimensions to the problem would probably mean that the points of the existing design had those new dimensions fixed at a certain level. This breaks the non-collapsing property of the Latin hypercube designs (or any other design for that matter). In reality, one would have to run new simulations in order to cover the new added dimensions. For situations like this, the best practice is to use optimization to guide where to place the new design points (see Table II for a discussion on sequential sampling).

3.2. How far has optimization of the Latin hypercube design gone?

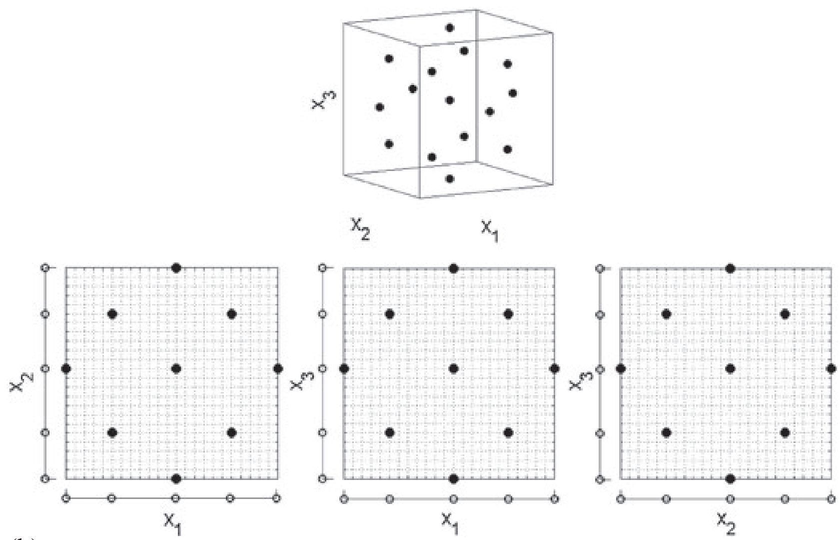
The need of flexible experimental design strategies took research in Latin hypercube design away from model-specific figures of merit (such as in D-optimal designs, which maximizes the determinant of the information matrix). For many applications, very little was assumed about the relationship between input and outputs (although covariance among inputs was often a topic such as when Latin hypercube sampling is used for multivariate integration). Instead, the figures of merit were expressed in terms of the input space (e.g., the minimum distance between points). The optimization of the Latin hypercube design has shown to be a challenging task because it is a combinatorial optimization problem with search space of the order of $(p\,!)^d$. For example, to optimize the location of 20 samples in two dimensions, the algorithm has to select the best design from more than 10^{36} possible designs. If the number of variables is

§*After all, only so much could be achieved if the modeling approach remained the same (i.e., polynomial response surfaces). Techniques such as Gaussian process (and kriging), radial basis function, neural networks, support vector machines, and others play an important role* [25–29] *as they can be used to flexibly model nonlinear functions. The Latin hypercube being model independent supports the use of all these modeling techniques.*

¶*The low dimensionality sometimes found in physical experiments is usually an imposition of the high cost associated with them (i.e., exploration of large design spaces is often prohibitively expensive).*



(a) Latin hypercube design with 15 points.



(b) Central composite design with 15 points.

Figure 3. One-dimensional and two-dimensional projections of 3-dimensional Latin hypercube and central composite designs

increased to three, the number of possible designs is more than 10^{55}. Another challenge is the objective function computation that tends to be time consuming. Most objective functions require the computation of a form of distance among the points in the design, which makes the number of operations to grow very fast with the number of points (and consequently number of variables because large dimensional spaces usually require large number of points).

The result is an abundant literature on optimization of the Latin hypercube point location [30–44] (just to cite a few, but the list could go on and on). Table I summarizes some strategies found in the literature. Overall, two research focus are recurrent, namely, the optimization algorithm and the objective function. In few cases, researchers explored both simultaneously. The interested reader can find databases of optimized Latin hypercube designs that can be downloaded from websites like http://www.spacefillingdesigns.nl and http://harvest.nps.edu.

In terms of optimization algorithms, coordinate exchange, columnwise-pairwise, and enhanced stochastic evolutionary algorithms are naturally suitable for Latin hypercube optimization. That is because they can deal with combinatorial problems, and they were designed to account for the non-collapsing structure of the Latin hypercube designs. However, literature has also shown success in the use of variations of discrete and continuous optimization methods such as simulated annealing, genetic algorithms, and particle swarm optimization.

As for the objective function, a lot of authors advocate in favor of the criteria intended to space-filling designs (e.g., potential energy, entropy, and ϕ_p – which is sometimes seen as a variant of the maximin distance criterion). Some authors argue that reduced correlation among inputs is also important (e.g., L2-discrepancy). Unfortunately, it is difficult to relate any of these criteria with the accuracy of the resulting metamodels. Probably because of that, the choice of objective function for Latin hypercube optimization is not unanimous. Nevertheless, some combinations of objective functions and algorithms might favor the overall performance of

Table I. Review of approaches for constructing the optimal Latin hypercube design (adapted from [40])

Researchers	Year	Algorithm	Objective functions
Audze and Eglajs [30]	1977	Coordinates exchange	Potential energy
Park [31]	1994	2-stage: exchange-type and Newton-type	Integrated mean-squared error and entropy criteria
Morris and Mitchell [32]	1995	Simulated annealing	ϕ_p criterion
Ye *et al.* [33]	2000	Columnwise-pairwise	ϕ_p and entropy criteria
Fang *et al.* [34]	2002	Threshold accepting	Centered L2-discrepancy
Bates *et al.* [35]	2004	Genetic algorithm	Potential energy
Jin *et al.* [36]	2005	Enhanced stochastic evolutionary algorithm	ϕ_p criterion, entropy, and L2 discrepancy
Liefvendahl and Stocki [37]	2006	Columnwise-pairwise and genetic algorithms	Minimum distance and potential energy
van Dam *et al.* [38]*	2007	Branch-and-bound	2-norm
Grosso *et al.* [39]	2008	Iterated local search and simulated annealing algorithms	ϕ_p criterion
Viana *et al.* [40]	2010	Translational propagation	ϕ_p criterion
Roustant *et al.* [41]	2010	No particular algorithm (focus on problem formulation)	Radial scanning statistic
Husslage *et al.* [42]	2010	Periodic designs and the enhanced stochastic evolutionary algorithm	maximin and Audze-Eglais
Zhu *et al.* [43]	2012	Successive local enumeration	Potential energy
Chen *et al.* [44]	2013	Particle swarm optimization	Max-min criterion

*They also show a construction method for optimized 1-norm and infinity-norm.

the optimization strategy. For example, Jin *et al.* [36] proposed strategies for efficient calculation of the ϕ_p, entropy, and L2 discrepancy criteria based on the fact that they were using the enhanced stochastic evolutionary algorithm. The ϕ_p, entropy, and L2 discrepancy calculations take advantage of the fact that only two elements in the design matrix are involved in each iteration of their algorithm (resulting in significant savings in computation time).

The growing power for computers changed the perception about the performance and limitation of the algorithms. For example, in 2000, Ye *et al.* [33] reported that generating an optimal Latin hypercube of 25 points and four variables would take several hours on a Sun SPARC 20 workstation. In 2005, Jin *et al.* [36] reported that it would take only 2.5 s to optimize the same size of design using a personal computer powered by a Pentium III 650 MHZ CPU. Today, time-consuming designs might be on the order of hundreds of points and/or several tens of variables.

Optimization and computer power are definitely strong drivers for the Latin hypercube popularity. The combination of these two elements has enabled experimental designs with very good space-filling properties at reasonable computational cost. Figure 4 shows the difference in number of papers acknowledging the use of Latin hypercube versus Hammersley designs over the years. The difference used to be small and favoring Hammersley designs. However, the trend started to change as the optimization strategies for Latin hypercube became mature, and implementations became fast enough from the user's perspective. As a warning, this does not mean that Latin hypercube are better than Hammersley designs. It only gives a hint that affordably optimized Latin hypercube designs tend to be used more often.

3.3. Do Latin hypercube designs have any drawback?

Most certainly, Latin hypercube designs have drawbacks. By virtue of their definition, Latin hypercube designs have good uniformity with respect to each dimension individually. Desirable properties, such as space filling, column-wise orthogonality, or conversely, designs with statistically dependent variables come at the cost of very expensive optimization. Unfortunately, these properties are not invariant under transformations. This is particularly important for applications where sampling is performed in one coordinate system, but modeling is performed in another (e.g., it is common to see engineering problems where the inputs of the computer code are mapped to a conveniently normalized space where physical interpretation of results becomes easier).

As discussed previously, there are cases where some input variables can be disregarded because they do not influence the output as much. If the design space was originally sample using Latin hypercube sampling, the resulting design after the dimensionality reduction is still a Latin hypercube design. Unfortunately, in such cases, there is no guarantee of optimality with respect to space filling (or any other figure of merit used in the optimization of the initial design). Figure 3(a) illustrates the case of an approximation of an optimized Latin hypercube design (obtained with the translational propagation Latin hypercube design algorithm proposed by Viana *et al.*)[40] with $d=3$ dimensions and $p=24$ points. Although any of the two-dimensional projections is still a Latin hypercube design with the same $p=24$ points, the space filling properties are hugely affected specially when x_1 is dropped out. Although the projection in x1-x2 is a much better representation of a space filling design than the other two projections, the distribution of points over all three two-dimensional projections seems to follow a diagonal pattern – same is observed in Figure 2(c). The interested reader is referred to Roustant *et al.* [41] for a discussion on how to minimize that effect.

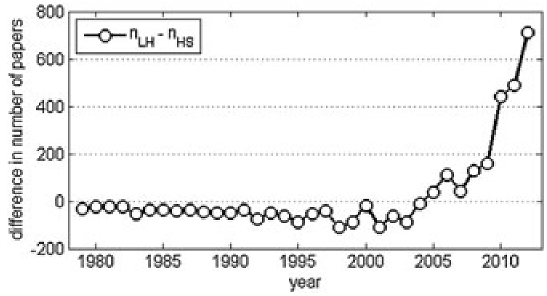


Figure 4. Difference between number of publications over the years. n_{LH} and n_{HS} are the number of papers using Latin hypercube and Hammersley designs according to Figure 1(a), respectively

In addition, improved space-filling properties may be achieved by minimizing some form of distance measure, whereas orthogonality may be obtained by considering column-wise correlations. By the way, orthogonality of the sampling points [45,46] is also observed in other sampling techniques such as orthogonal arrays. Unfortunately, literature [47–53] has shown that optimization with respect to either of these properties does not necessarily lead to the best experimental design with respect to the other property. Figure 5 shows the scatter plot of the column-wise correlation and ϕ_p criteria when 1000 two-dimensional Latin hypercube design 20 points are created with the MATLAB function *lhsdesign* [54] (using default parameters). This figure illustrates how difficult it might be to find the experimental design that minimizes both the column-wise correlation and ϕ_p criteria. In addition, it is clear that the best designs for the column-wise correlation can greatly differ in terms of ϕ_p criterion and vice versa. The work by Hernandez *et al.* [53] and MacCalman [55] discuss in depth the minimization of pairwise correlation. The interested reader can find their algorithm

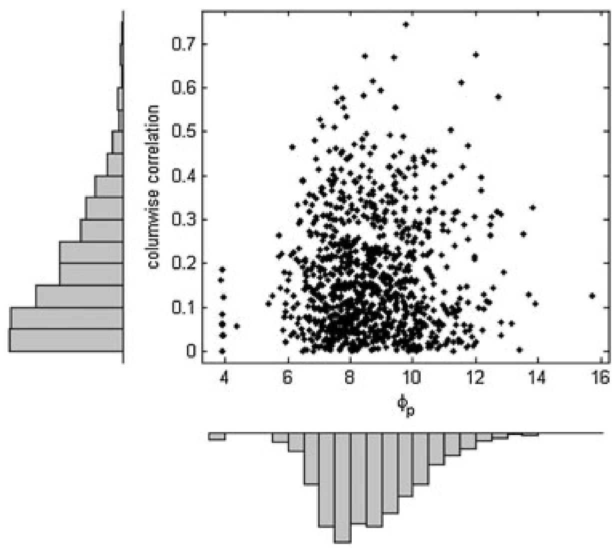


Figure 5. Scatterplot of column-wise correlation and ϕ_p criteria out of 1000 Latin hypercube designs with $d = 2$ dimensions and $p = 20$ points created with the MATLAB function *lhsdesign* [54]

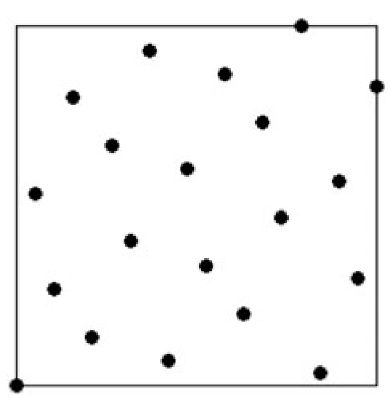


Figure 6. Example of Hammersley design with $d = 2$ dimensions and $p = 20$ points

Table II. Few suggestions of research topics

Topic	Comments and examples of work being performed
Optimization of Latin hypercube	Comment: the growing in computer power has made optimization feasible from small to moderate sampling sizes. However, creating experimental designs in large dimensional spaces (when the sample size is also naturally large) is still very challenging. The task is time consuming because computation of optimization criteria and number of iterations for convergence (and there is always the debate around the curse of dimensionality). Example of work: Owen [61] introduced Latin supercube sampling (combination of Latin hypercube with Quasi Monte Carlo) for numerical integration of functions defined in very high dimensional spaces.
Mixing discrete and continuous variables	Comment: because each dimension is divided into equal number of levels (which is also equal to the number of points in the experimental design), Latin hypercube designs face limitations when dealing with discrete variables. The number of levels may not coincide with the levels of the discrete variables. One can always map few Latin hypercube levels into a single level of the discrete variable (which might end up being a sub-optimum strategy). Example of work: Meckesheimer *et al.* [62] reviewed different sampling and surrogate techniques when applied to discrete/continuous problems.
Incorporation of global sensitivity information	Comment: it can happen that one knows beforehand that the response varies much slower in certain dimensions (or faster in others). Think about sampling a model that is known to be linear in one dimension and nonlinear in another dimension (e.g., $y = x_1 + x_2 + \cos(x_2) + x_2^2$). It is possible to conceive that two or three levels in x1 should be enough, while $x2$ might need more than that. If 10 sample points are available, the traditional formulation of the Latin hypercube would create 10 levels in each dimension. It would be interesting to change the formulation such that the sensitivity information could be incorporated and potentially help allocating better the sampling points. In such cases, it would make sense to have few levels in the slower changing dimensions and more levels in the fast changing ones. Example of work: researchers like Sobol [63] and Tarantola *et al.* [64] have proposed strategies for assessing global sensitivities. Nevertheless, to the best of my knowledge, there is no sampling strategy that would use existing sensitivity information (i.e., without relying on model) to decide where to conduct simulations/experiments.
Constrained design space	Comment: it is relatively common to find problems where outputs are not feasible (physically or numerically) throughout the hypercube defined by lower and upper bounds of input variables. In such cases, it might be beneficial to constraint the design of experiment, which imposes a layer of complexity to its optimization. Example of work: Petelet *et al.* [65] introduced a technique that creates Latin hypercube sample taking into account inequality constraints among input variables. The technique relies on optimized permutations of an initial Latin hypercube design. Similarly, Bowman and Woods [66] proposed a method that optimizes space-filling designs, weighting the distance between points by multivariate dependencies between input variables.
Saturated and supersaturated designs	Comment: as discussed in the recent reviews,[67,68] there are cases, such as screening, where a relatively large number of input variables are studied. When there is strong limitation on the size of the experimental design, then saturated and supersaturated designs may have to be considered. In such designs, the number of input variables or degrees of freedom in the modeling technique is close to or exceeds the number of runs. This is a very important topic in the design and analysis of computer experiments because the computational costs associated with the simulations (e.g., in computational fluid dynamics and crash analysis) can drastically limit the size of the experimental designs (not only used for screening but also in the actual modeling itself). Example of work: Butler [69] presented a discussion on supersaturated Latin hypercube designs and the conditions for a design to be optimal under the $E(s^2)$ -optimality criterion (used in supersaturated designs).
Sequential sampling	Comment: this is the case in which a first set of points is used to create a surrogate model, which turns out to perform poorly. Then, the experimental design is augmented so that with the new points the quality of the surrogate model would improve. Example of work: Rennen *et al.* [70] proposed nested designs, which the resulting new design is also a Latin hypercube (one advantage of such approach is the easy of computation and potentially large number of new points). On the other hand, several authors have proposed using surrogate models, such as the Gaussian process (e.g., by maximizing its uncertainty) and ensemble of surrogates (e.g., by maximizing a disagreement metric), to help identifying the next sampling points. The interested reader is referred to [17] and [71–75].

freely available at the Simulation Experiments and Efficient Designs (SEED) Center website (http://harvest.nps.edu), in addition to a catalogue of ready-to-use, nearly orthogonal designs for up to 22 variables in as few as 129 points.

As any other design of experiment, the Latin hypercube also suffers from the curse of dimensionality.[56] While uniformity in each dimension is preserved, the space filling properties become questionable. As the number of variables increase, it becomes harder to fill the design space. When optimization pushes points further apart, the sample tends to create a vacuum in the center of the design space. Again, this is not observed only in Latin hypercube designs, and it is something that is typical in high dimensions.

3.4. When studying computer experiments, should Latin hypercube designs always be used?

In general, design for computer experiments should observe good coverage of the design space, many levels for each variable, and good projection properties. Sure, the Latin hypercube designs conveniently suit all that, but that does not guarantee they will always provide the best initial sample for a given problem. Figure 6 illustrates a Hammersley design in two-dimensional space. Using only visual inspection, this design could very well be rated as a Latin hypercube with good space filling properties, just like the one illustrated in Figure 2(c).

Because different sampling strategies (including their optimized variants) might lead to similar designs in terms of space-filling properties, literature does not point to any particular approach that would work best all the time.[57,58] In fact, Qu and Haftka [59] and Goel *et al.* [60] have empirically demonstrated losses associated with using a single experimental design strategy (they even suggested combing different design of experiments, such as full factorial and face centered with the Latin hypercube design, when the computational budget allows).

3.5. Where are the research opportunities in Latin hypercube design?

Although a lot was accomplished since the introductory papers of Mckay *et al.* [18] and Iman and Conover,[19] there are still plenty of open issues. Table II discusses some research topics that are still open. Overall, one strong tendency is to expand the capabilities by possibly having to relax some of the Latin hypercube properties.

4. Summary and conclusions

This paper presented an overview of the research in Latin hypercube sampling. The objective was to offer, especially to the newcomers, a critical view on this class of design of experiments. To do that, the discussion was divided into five topics: popularity, optimization, drawbacks, alternatives, and open issues. After going through these topics, the reader should have an appreciation for the work that was already performed to improve the originally proposed Latin hypercube design.

The reader should be able to recognize that Latin hypercube designs are very well accepted (particularly in studying computer experiments) because of flexibility in terms of data density and location, and in addition, non-collapsing and optimizable space-filling properties. Hopefully, this paper made the research in terms of Latin hypercube optimization to be better appreciated both in terms of algorithm and problem formulation.

The reader should also have a clear perception on the limitation of Latin hypercube sampling. The bottom line is that there is still research on optimization schemes that generates space-filling and uncorrelated samples, as well as how to extend Latin hypercube designs to large dimensional spaces. With that in mind, the reader was pointed to some papers about experimental designs for computer experiments that are alternatives to Latin hypercube sampling (e.g., orthogonal arrays and Hammersley designs). This paper also mentions literature that discusses the combination of sampling strategies when developing surrogate models.

Finally, some open issues and opportunities were highlighted in Table II. They should not be understood as the only important research topics in Latin hypercube sampling. Instead, they represent a collection of relevant themes from the perspective of the engineering design and optimization community. The presented discussion hopefully raised awareness about capabilities, limitations, and what remains to be performed while providing a collection of relevant references for the interested reader.

Acknowledgements

The author is very thankful to GE for supporting the publication of this paper. The views expressed here reflect the views of the author alone and do not necessarily reflect the views of GE.

References

1. J Sacks, WJ Welch, TJ Mitchell, HP Wynn, “Design and analysis of computer experiments,” *Statistical Science* 1989; **4**, 409–435.
2. TJ Santner, BJ Williams, WI Notz, The design and analysis of computer experiments, Springer Verlag: New York, USA, 2003. ISBN 0-387-95420-1.
3. FAC Viana, TW Simpson, V Balabanov, V Toropov, “Metamodeling in multidisciplinary design optimization: how far we have come?,” *AIAA Journal* 2014; **52** (4), 670–690. doi: 10.2514/1.J052375
4. AIJ Forrester, AJ Keane, “Recent advances in surrogate-based optimization,”*Progress in Aerospace Sciences* 2009; **45** (1–3), 50–79.
5. Kleijnen JPC. Design and Analysis of Simulation Experiments. Springer: New York, 2009.

6. K. Yu, X. Yang, Z. Yue, “Aerodynamic and heat transfer design optimization of internally cooling turbine blade based different surrogate models,”*Structural and Multidisciplinary Optimization* 2011; **44** (1): 75–83.
7. FAC Viana, J Madelone, N Pai, G Khan, S Baik, “Temperature-based optimization of film cooling in gas turbine hot gas path components,” *ASME Turbo Expo* 2013, San Antonio, USA, Jun **03–07**: GT2013–94293.
8. J. Eves, V.V. Toropov, H.M. Thompson, N. Kapur, J. Fan, D. Copley, A. Mincher, “Design optimization of supersonic jet pumps using high fidelity flow analysis,” *Structural and Multidisciplinary Optimization* 2012; **45** (5): 739–745.
9. R. Moshfegh, L. Nilsson, M. Larsson, “Estimation of process parameter variations in a pre-defined process window using a Latin hypercube method,” *Structural and Multidisciplinary Optimization* 2008; **35** (6): 587–600.
10. AL Marsden, AF Jeffrey, AT Charles, “A computational framework for derivative-free optimization of cardiovascular geometries,” *Computer Methods in Applied Mechanics and Engineering* 2008; **197**(21): 1890–1905.
11. C Dopico-González, AM New, M Browne, “Probabilistic analysis of an uncemented total hip replacement,” *Medical Engineering & Physics* 2009; **31**(4): 470–476.
12. JPC Kleijnen, H Pierreval, J Zhang, “Methodology for determining the acceptability of system designs in uncertain environments,” *European Journal of Operational Research* 2011; **209** (2): 176–183.
13. MD Morris, TJ Mitchel, “Exploratory designs for computer experiments,” *Journal of Statistical Planning and Inference* 1995; **43**: 381–402.
14. TW Simpson, DKJ Lin, W Chen, “Sampling strategies for computer experiments: design and analysis,” *International Journal of Reliability and Applications* 2001; **2** (3): 209–240.
15. KT Fang, R Li, “Uniform design for computer experiments and its optimal properties,” *International Journal of Materials and Product Technology* 2006; **25** (1), 198–210.
16. VCP Chen, KL Tsui, RR Barton, M Meckesheimer, “A review on design, modeling and applications of computer experiments,” *AIIE Transactions, Vol* 2006; **38** (4):273–291.
17. L Pronzato, WG Müller, “Design of computer experiments: space filling and beyond,” *Statistics and Computing* 2012; **22**(3), 681–701.
18. MD Mckay, RJ Beckman, WJ Conover, “A comparison of three methods for selecting values of input variables in the analysis of output from a computer code,” *Technometrics* 1979; **21** (2); 239–245.
19. RL Iman, WJ Conover, “Small sample sensitivity analysis techniques for computer models, with an application to risk assessment,” *Communications in Statistics Part A. Theory and Methods*, 1980; **17**, 1749–1842.
20. AB Owen, “Orthogonal arrays for computer experiments, integration and visualization,” *Statistics Sinica* 1992; **2**, 439–452.
21. UM Diwekar, JR Kalagnanam, “Efficient sampling technique for optimization under uncertainty,” *AIChE Journal* 1997; **43**(2): 440–447.
22. JM Hammersley, “Monte Carlo methods for solving multivariate problems,” *Annals of the New York Academy of Sciences*1960; **86**(3): 844–874.
23. Koziel S, Leifsson L (Eds)(ed). Surrogate-based modeling and optimization – Applications in engineering. Springer: New York, 2013.
24. AA Giunta, SF Wojtkiewicz Jr, MS Eldred, “Overview of modern design of experiments methods for computational simulations,” *41st AIAAAerospace Sciences Meeting and Exhibit* 2003 Reno, Nevada, Jan **6–9**: AIAA 2003–0649.
25. CE Rasmussen, CKI Williams. Gaussian Processes for Machine Learning. The MIT Press: Cambridge, USA, 2006. ISBN 0-262-18253-X.
26. Stein ML. Interpolation of Spatial Data: Some Theory for Kriging. Springer Verlag, 1999.
27. J Park, IW Sandberg, “Universal approximation using radial-basis-function networks,” *Neural Computation* 1991; **3**(2): 246–257.
28. M Smith, Neural networks for statistical modeling, Von Nostrand Reinhold, 1993.
29. B Schölkopf, AJ Smola. Learning with Kernels. The MIT Press: Cambridge, USA, 2002. ISBN 9780262194754.
30. P Audze, V Eglajs, “New approach for planning out of experiments,” *Problems of Dynamics and Strengths* 1977; **35**; 104–107 (in Russian).
31. JS Park, “Optimal Latin-hypercube designs for computer experiments,” *Journal of Statistical Planning and Inference* 1994; **39**: 95–111.
32. MD Morris, TJ Mitchell. “Exploratory designs for computational experiments,” *Journal of Statistical Planning and Inference* 1995; **43**: 381–402.
33. KQ Ye, W Li, A Sudjianto. “Algorithmic construction of optimal symmetric Latin hypercube designs,” *Journal of Statistical Planning and Inference* 2000; **90**: 145–159.
34. Fang KT, Ma CX, Winker P. Centered L2-discrepancy of random sampling and Latin hypercube design and construction of uniform designs,”. *Mathematics of Computation* 2002; **71**:275–296.
35. SJ Bates, J Sienz, VV Toropov, “Formulation of the optimal Latin hypercube design of experiments using a permutation genetic algorithm” *45th AIAA/ASME/ASCE/AHS/ASC Structures Structural Dynamics and Materials Conference, Palm Springs, CA, April* 19–22, 2004.
36. R Jin, W Chen, A Sudjianto, “An efficient algorithm for constructing optimal design of computer experiments,” *Journal of Statistical Planning and Inference* 2005; **134**, 268–287.
37. M Liefvendahl, R Stocki “A study on algorithms for optimization of Latin hypercubes,” *Journal of Statistical Planning and Inference* 2006; **136**(9): 3231–3247.
38. E van Dam, B Husslage, D den Hertog, H Melissen, “Maximin Latin hypercube designs in two dimensions,” *Operations Research* 2007; **55**(1): 158–169.
39. A Grosso, A Jamali, M Locatelli, “Finding maximin Latin hypercube designs by iterated local search heuristics,” *European Journal of Operational Research* 2009; **197**(2): 541–547.
40. FAC Viana, G Venter, V Balabanov, “An algorithm for fast generation of optimal Latin hypercube designs,” *International Journal for Numerical Methods in Engineering* 2010; **82**: 135–156.
41. O Roustant, J Franco, L Carraro, A Jourdan, “A radial scanning statistic for selecting space-filling designs in computer experiments,” In *mODa 9 – Advances in Model-Oriented Design and Analysis*, Physica-Verlag HD, 189–196, 2010.
42. BGM Husslage, G Rennen, ER van Dam, D den Hertog, “Space-filling Latin hypercube designs for computer experiments,” *Optimization and Engineering* 2011; **12**(4): 611–630.
43. H Zhu, L Liu, T Long, L Peng, “A novel algorithm of maximin Latin hypercube design using successive local enumeration,” *Engineering Optimization* **44** (5): 2012; 551–564.
44. RB Chen, DN Hsieh, Y Hung, W Wang, “Optimizing Latin hypercube designs by particle swarm,” *Statistics and Computing* 2013; **23** (5): 663–676.
45. TW Simpson, JD Peplinski, PN Koch, JK Allen, “Meta-models for computer based engineering design: survey and recommendations,” *Engineering with Computers* 2001; **17**(2): 129–150.
46. JPC Kleijnen, SM Sanchez, TW Lucas, TM Cioppa, “A user’s guide to the brave new world of designing simulation experiments,” *INFORMS Journal on Computing* 2005; **17**(3); 263–289.
47. M Stein, “Large sample properties of simulations using Latin hypercube sampling,” *Technometrics* 1987; **29** (2): 143–151.
48. KQ Ye, “Orthogonal column Latin hypercubes and their application in computer experiments,” *Journal of the American Statistical Association* 1998; **93**(444): 1430–1439.
49. M Liefvendahl, R Stocki, “A study on algorithms for optimization of Latin hypercubes,” *Journal of Statistical Planning and Inference* 2006; **2006**: 3231–3247.
50. TM Cioppa, TW Lucas, “Efficient nearly orthogonal and space-filling Latin hypercubes,” *Technometrics* 2007; **49**(1): 45–55.
51. P Prescott, “Orthogonal-column Latin hypercube designs with small samples,” *Computational Statistics and Data Analysis* 2009; **53**: 1191–1200.

52. F Sun, MQ Liu, DKJ Lin, "Construction of orthogonal Latin hypercube designs with flexible run sizes," *Journal of Statistical Planning and Inference* 2010; **140**: 3236–3242.
53. AS Hernandez, TW Lucas, M Carlyle. "Constructing nearly orthogonal Latin hypercubes for any nonsaturated run-variable combination," *ACM Transactions on Modeling and Computer Simulation* 2012; **22**(4): article 20.
54. Mathworks contributors, MATLAB: the language of technical computing, Version 7.12.0.635 (R2011a), The MathWorks Inc., 2011.
55. AD MacCalman, Flexible space-filling designs for complex system simulations, PhD Dissertation, Naval Postgraduate School, Monterey, CA, USA, June, 2013.
56. van Dam ER, Rennen G, Husslage B. Bounds for maximin Latin hypercube designs. *Operations Research* 2009; **57**(3):595–608.
57. SB Crary, "Design of computer experiments for metamodel generation," *Analog Integrated Circuits and Signal Processing, Vol* 2002; **32**(1): 7–16.
58. D Bursztyn, DM Steinberg, "Comparison of design for computer experiments," *Journal of Statistical Planning and Inference* 2006; **136**: 1103–1119.
59. X Qu, RT Haftka, "Reliability-based design optimization using probability sufficiency factor," *Structural and Multidisciplinary Optimization* 2004; **27**(5): 314–325.
60. T Goel, RT Haftka, W Shyy, LT Watson, "Pitfalls of using a single criterion for selecting experimental designs," *International Journal for Numerical Methods in Engineering* 2007; **75**: 127–155.
61. AB Owen, "Latin supercube sampling for very high-dimensional simulations," *ACM Transactions on Modeling and Computer Simulation* 1998; **8**(1): 71–102.
62. M Meckesheimer, RR Barton, TW Simpson, F Limayem, B Yannou, "Metamodeling of combined discrete/continuous responses," *AIAA Journal* 2001; **39**(10): 1950–1959.
63. IM Sobol, "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates," *Mathematics and Computers in Simulation* 2001; **55**(1–3): 271–280.
64. S Tarantola, W Becker, D Zeitz, "A comparison of two sampling methods for global sensitivity analysis," *Computer Physics Communications* 2012; **183** (5): 1061–1072.
65. M Petelet, B Iooss, O Asserin, A Loredo, "Latin hypercube sampling with inequality constraints," *AStA Advances in Statistical Analysis* 2010; **94**(4): 325–339.
66. VE Bowman, DC Woods, "Weighted space-filling designs," *Journal of Simulation*, available online, doi: 10.1057/jos.2013.8.
67. CJ Marley, DC Woods, "A comparison of design and model selection methods for supersaturated experiments," *Computational Statistics & Data Analysis* 2010; **54**(12): 3158–3167.
68. SD Georgiou, "Supersaturated designs: a review of their construction and analysis," *Journal of Statistical Planning and Inference*, Available online, doi:10.1016/j.jspi.2012.09.014.
69. NA Butler, "Supersaturated Latin hypercube designs," *Communications in Statistics - Theory and Methods* 2005; **34**(2): 417–428.
70. G Rennen, B Husslage, ER van Dam, D den Hertog, "Nested maximin Latin hypercube designs," *Structural and Multidisciplinary Optimization* 2010; **41** (3): 371–395.
71. Jones DR, Schonlau M, Welch WJ. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization* 1998; **13** (4):455–492.
72. F Xiong, Y Xiong, W Chen, S Yang, "Optimizing Latin hypercube design for sequential sampling of computer experiments," *Engineering Optimization* 2009; **41** 793–810.
73. K Crombecq, E Laermans, T Dhaene, "Efficient space-filling and non-collapsing sequential design strategies for simulation-based modeling," *European Journal of Operational Research* 2011; **214**: 683–696.
74. Goel T, Haftka RT, Shyy W, Queipo NV. Ensemble of surrogates. *Structural and Multidisciplinary Optimization* 2007; **33**(3):199–216.
75. F.A.C. Viana, RT Haftka, LT Watson "Efficient global optimization algorithm assisted by multiple surrogate techniques," *Journal of Global Optimization* 2013; **56**(2): 669–689.
76. RA Fisher, The design of experiments, Oliver and Boyd, 1935.
77. F Yates, "The principles of orthogonality and confounding in replicated experiments," *The Journal of Agricultural Science* 1933; **23**: 108–145.
78. ES Pearson, "Some aspects of the problem of randomization," *Biometrika* 1937; **29**(1–2); 53–64.
79. F Yates, WG Cochran, "The analysis of groups of experiments," *The Journal of Agricultural Science* 1938; **28**: 556–580.
80. RS Crutchfield, "Efficient factorial design and analysis of variance illustrated in psychological experimentation," *The Journal of Psychology* 1938; **5**(2): 339–346.
81. P Ifju, D Jenkins, S Ettinger, Y Lian, W Shyy, M Waszak, "Flexible-wing-based micro air vehicles," *40th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, USA, Jan 14–17, 2002, AIAA 2002-0705.
82. BD Ripley. Stochastic Simulation. John Wiley & Sons, Inc.: New York, USA, 2008. ISBN 9780471818847.
83. AM Law. Simulation Modeling and Analysis (5th edn). McGraw-Hill Education: New York, USA, 2013. ISBN-13: 978-0073401324.

Appendix: Design and analysis of physical versus computer experiments

One might be tempted to see more similarities than differences between design and analysis of physical and computer experiments. After all, the goal is to build a model that represents the relationship between inputs and outputs at a satisfactory degree of accuracy. Certainly, design and analysis of physical experiments is a much older field (for example, see the seminal book by Fisher [76] or early papers like).[77–80] Thus, sampling and modeling techniques for physical experiments were first developed in a time where computers had very limited capabilities (that might be a reason for the extensive use of polynomial response surface). On the other hand, the design and analysis of computer experiments is relatively recent (see seminal paper by Sacks *et al.*);[1] and since the beginning, the field could take advantage of better computers.

Within the context of this paper, computer models are deterministic[‖] (given a set of inputs, the outputs are always the same) and build to represent physical systems at given fidelity level (degree of physics the model can describe). On the other hand, physical experiments are stochastic because they are susceptible to measurement error and nuisance variables (that may or may not be recognized) causing variation in the response. The notion of fidelity level may appear, but physical experiments tend to be seem

[‖]*Obviously, there are stochastic computer codes (e.g., simulations based on Monte Carlo methods). The interested reader can find further reading in* [82,83].

as observation of reality (again, probably corrupted by some error). Obviously, there are cases (e.g., high-energy physics) where physical similitude needs to be explored. In such cases, an easier-to-control experiment 'approximates reality' (e.g., experiments conducted at lower temperatures with results appropriately scaled to estimate what would happen at high temperatures).

With that said, design of experiment strategies for physical experiments needs to incorporate concepts like randomization, blocking, and replication. Randomizing the order of applying the experimental inputs is important for mitigating the effect of unrecognized nuisance variables. Blocking (arrangement of experimental units in groups) reduces the impact of recognized nuisance variables; and finally, replication helps quantify and reduce the effects of measurement error. Obviously, these concepts are not as important (or not at all) in computer experiments (there are no nuisance variables and, in the deterministic case, no equivalent of measurement error).

In general, computer experiments are used to make analysis of physical systems easier (or sometimes possible). This notion implies that computer experiments tend to be relatively cheaper than their physical counterparts.** As a result, computer models are used in sensitivity analysis, reliability assessment, design optimization, and a number of other studies that tend to require a large number of function evaluations. Very often, there is limited previous knowledge (particularly in situations like conceptual design) and engineers, designers, and analysts tend explore a large number of input variables (defined over relatively large domains). Obviously, physical experiments are still part of most system analysis. For one thing, data from physical experiments would be used for validating, calibrating, and quantifying uncertainty in computer models. In engineering systems, physical experiments are vital in early phases of design such as in material characterization, coupon, and element tests.

Authors' biography

Felipe A. C. Viana is a member of the Probabilistics Laboratory at GE Global Research. His responsibilities include delivering state-of-the-art probabilistic design methods with applications in aero-thermal and mechanical systems from new designs to fielded product quality improvement across GE businesses. He earned his first Doctor of Philosophy degree in Mechanical Engineering from the Federal University of Uberlandia (Brazil) with the dissertation “Surrogate Modeling Techniques and Heuristic Optimization Methods Applied to Design and Identification Problems” in 2008. He earned a second doctorate, in Aerospace Engineering, from the University of Florida with the dissertation “Multiple Surrogates for Prediction and Optimization” in 2011. The research of Dr. Viana has produced more than 50 papers in the top journals and conferences in the field of probabilistics and optimization methods. His research interests include design and analysis of experiments, probabilistic modeling and uncertainty quantification, Bayesian statistics, and multidisciplinary design optimization.

[**]Keep in mind that there might be cases in which that does not hold. For example, studying flapping wings in micro aerial vehicles [81] might sometimes be cheaper through prototyping rather than simulations. Once the test facility is available (which tends to be the expensive part), building and testing different wing configurations become more affordable than running full-fledged unsteady computational fluid dynamics models.